

## **BAB IV**

### **DESKRIPSI PEKERJAAN**

Aplikasi *Monitoring Network* yang dijalankan di Dinas Pendapatan Provinsi Jawa Timur sudah menggunakan sebuah aplikasi. Disini saya hanya memperbaiki aplikasi yang ada disana dengan menambahkan dua filter yaitu dapat menampilkan kapasitas *bandwidth* di setiap *Server* dan menampilkan waktu terakhir *online* dan *offline* di setiap *Server*.

Kerja praktek ini dilakukan dengan pembagian waktu dalam satu minggu dalam 2 hari yang dilakukan selama 2 bulan, 8 jam setiap hari. Dalam kerja praktek ini, diharuskan menemukan permasalahan yang ada, mempelajari serta memberikan solusi bagi masalah yang timbul. Untuk mengatasi permasalahan tersebut, diperlukan langkah-langka yaitu:

- a. Menganalisa sistem.
- b. Mendesain sistem.
- c. Mengimplementasikan sistem.
- d. Melakukan pembahasan terhadap hasil implementasi sistem.

Keempat langkah tersebut dilakukan agar dapat menemukan solusi dari permasalahan yang ada. Lebih jelasnya dipaparkan pada *sub* bab dibawah ini.

#### **4.1 Menganalisa Sistem**

Menganalisa sistem merupakan langkah awal dalam membuat sistem baru. Langkah pertama adalah melakukan wawancara. Wawancara dilakukan oleh 1 orang dengan staff yang ditunjuk oleh Kepala Bagian Bidang Pengembangan

Sistem Informasi & Pengolahan *Data* Pendapatan. Staff tersebut memberikan informasi tentang mekanisme Aplikasi *Monitoring Network*.

Mekanisme Aplikasi *Monitoring Network* yang ada di Dinas Pendapatan Provinsi Jawa Timur hanya menampilkan status *connect* atau *disconnect Server* yang terdapat di seluruh kota se- Jawa Timur dengan penghubungannya jaringan internet.

Berdasarkan permintaan Kepala Bagian Bidang Pengembangan Sistem Informasi & Pengolahan *Data* Pendapatan, mengembangkan Aplikasi *Monitoring Network* dengan penambahan kapasitas *bandwidth* serta waktu *connect* atau *disconnect*.

#### **4.2 Mendesain Sistem**

Desain Sistem merupakan tahap pengembangan setelah analisis sistem dilakukan. Langkah-langka yang dilakukan dalam desain sistem ini adalah:

- a. *System Flow*
- b. *Context Diagram*
- c. *Data Flow Diagram*
- d. *Conceptual Data Model*
- e. *Physical Data Model*
- f. Struktur Tabel

*System flow* dibuat dengan menambahkan satu filter dari *system flow* yang lama. Selanjutnya membuat *context diagram*. *External Entity* dan proses-proses yang terjadi pada *context diagram* didapat dari *system flow* yang telah dibuat. Pendesain kemudian menyusun secara lengkap masing-masing proses

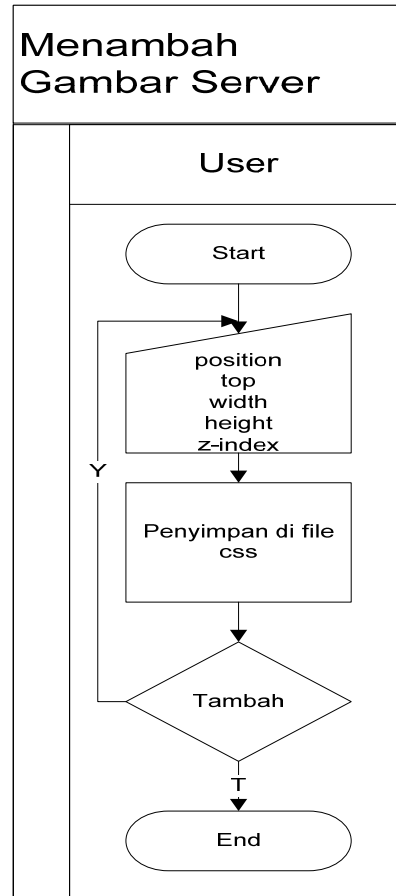
beserta *file-file* yang dibutuhkan pada *data flow diagram*. *File* yang terdapat pada *data flow diagram* digunakan sebagai acuan membuat *entity relational diagram* dan Struktur tabel.

#### 4.2.1 System Flow

Berikut ini merupakan sistem yang telah dikembangkan dari sistem yang ada. Dalam *system flow*, beberapa Proses tidak mengalami perubahan cuma menambahkan satu fitur baru.

##### a. Menambah Gambar *Server*

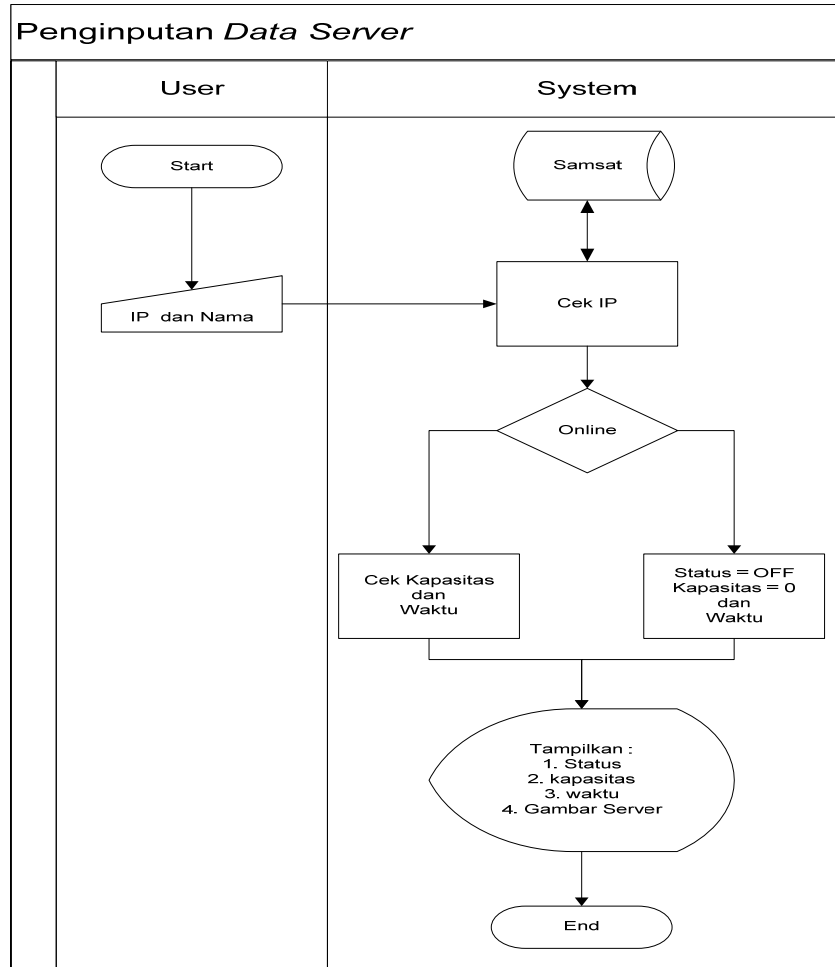
Pada proses ini *user* menginputkan *position*, *top*, *width*, *height* dan *z-index* yang terdapat di dalam *file* *css*. Setelah itu *user* menyimpan atau *save* kembali *file* *css* tersebut. Jika *user* ingin menambah gambar lagi berarti *user* harus menginputkan kembali *data-data* yang dibutuhkan tadi. Proses tersebut dapat dilihat pada Gambar 4.1 yang terdapat di bawah ini:



Gambar 4.1 *System Flow Menambah Gambar Server.*

b. Penginputan *Data Server*

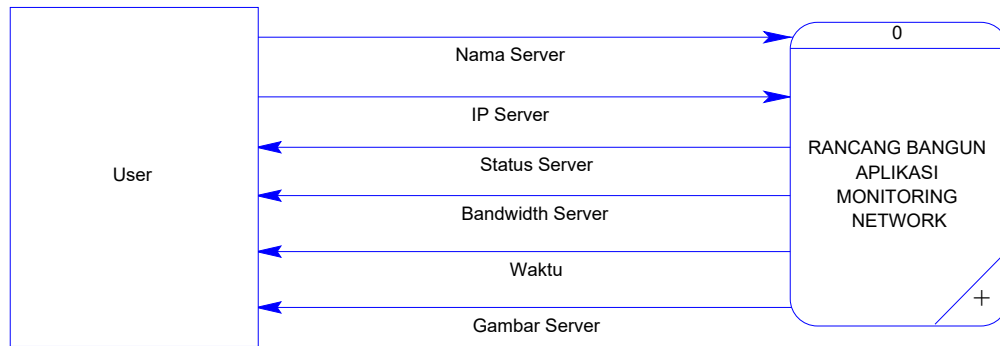
Pada proses ini *user* hanya menginputkan IP dan Nama *Server*, setelah itu sistem akan mengecek IP *Server* yang dimasukan apakah dalam status *connect* atau *disconnect*. Jika status *Server connect* akan menampilkan kapasitas *bandwidth* jaringan dalam ukuran Kbps dan jika status *disconnect* akan menampilkan kapasitas *bandwidth* jaringan 0 (nol) Kbps serta menampilkan waktu saat status *Server* dalam keadaan *connect* atau *disconnect*. Tidak hanya mengecek IP *Server* saja dalam proses ini juga menyimpan *data* yang diinputkan oleh *user* kedalam *database* yang bernama Samsat.



Gambar 4.2 System Flow Penginputan Data Server.

#### 4.2.2 Context Diagram

Pada *Context Diagram* Aplikasi *Monitoring Network* ini terdiri dari satu entitas (bagian), yaitu *user*. Satu entitas tadi memberikan inputan *data* yang berupa Nama *Server* dan IP *Server*. *user* tidak hanya menginputan saja tetapi juga menerima *output data* yang diperlukan seperti status *Server*, *bandwidth Server*, waktu dan gambar *Server*. Yang dapat dilihat di Gambar 4.3.



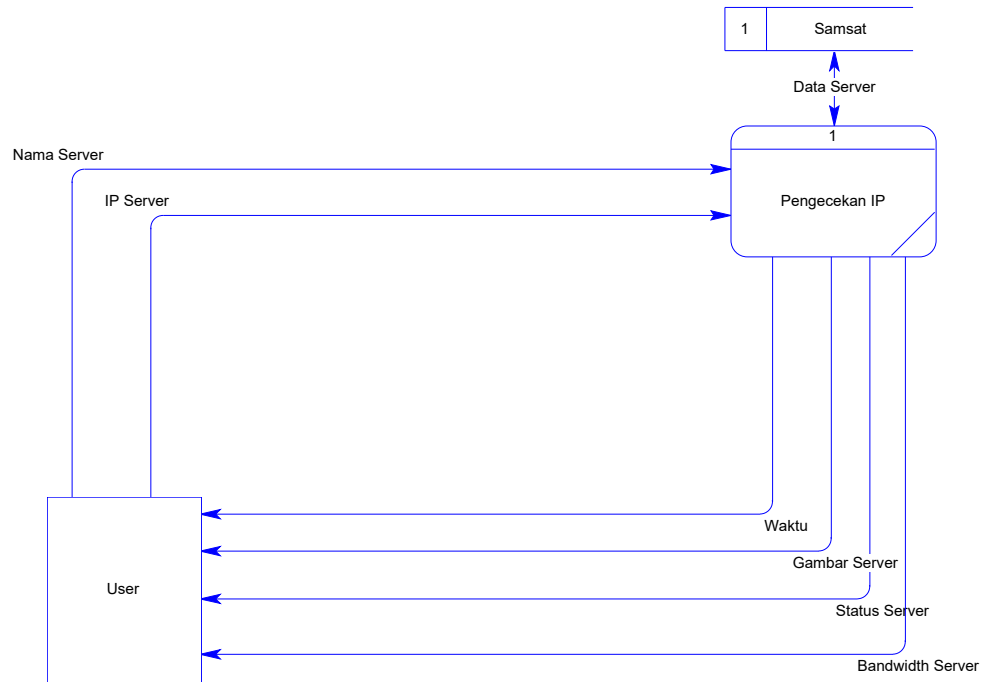
Gambar 4.3 *Context Diagram Aplikasi Monitoring Network.*

#### 4.2.3 Data Flow Diagram

Berikut ini adalah *Data Flow Diagram* Aplikasi *Monitoring Network*.

Dari sinilah kita bisa mengetahui aliran *data* yang ada didalam sistem yang terstruktur. Dalam perancangan Aplikasi *Monitoring Network* penggunaan *data flow diagram* adalah suatu cara yang mungkin harus digunakan, hal ini disebabkan supaya dapat mempermudah dalam memahami aplikasi yang ada dalam suatu perusahaan ataupun badan usaha.

*Data flow diagram* ini terdiri dari level 0. Dimana pada level 0 terdapat satu proses, yaitu proses pengecekan IP. Selain proses pengecekan IP tersebut terdapat satu *database* yaitu Samsat. Yang dapat dilihat pada Gambar 4.4.



Gambar 4.4 *Data Flow Diagram* Aplikasi *Monitoring Network*.

#### 4.2.4 Entity Relational Diagram (ERD)

*Entity Relational Diagram* (ERD) merupakan gambaran struktur *database* dari Aplikasi *Monitoring Jaringan* yang telah dikembangkan. *Entity Relational Diagram* (ERD) Dibagai menjadi dua, yaitu *Conceptual Data Model* (CDM) atau secara logik adan *Physical Data Model* (PDM) atau secara fisik.

##### a. *Conceptual Data Model* (CDM)

Pada *Conceptual Data Model*, terdapat 1 tabel yaitu table *samsat*. Yang berisikan id, ip, nama, status, kapasitas, waktu dan dex. Yang dapat dilihat di Gambar 4.5 yang terdapat di bawah ini.

Samsat
<u>id</u>
<u>ip</u>
nama
status
kapasitas
lineon
lineoff
dex

Gambar 4.5 *Conceptual Data Model Aplikasi Monitoring Network.*

b. *Physical Data Model (PDM)*

Pada *Physical Data Model* yang terdapat pada Gambar 4.6 dibawah ini, hanya tertera tipe *data* yang digunakan dari *conceptual data model* yang terdapat pada Gambar 4.5.

SAMSAT	
<u>ID</u>	<u>integer</u>
<u>IP</u>	<u>varchar(15)</u>
NAMA	varchar(10)
STATUS	varchar(10)
KAPASITAS	varchar(10)
LINEON	varchar(25)
LINEOFF	varchar(25)
DEX	varchar(10)

Gambar 4.6 *Physical Data Model Aplikasi Monitoring Network.*

#### 4.2.5 Struktur Tabel

Program Rancang Bangun Aplikasi *Monitoring Network* Berbasis *Web* pada Dispenda Jatim ini, hanya memiliki 1 *database* yaitu tabel Samsat. Struktur tabel Samsat sebagai berikut:

1. Tabel Samsat

Primary key : id, ip

Foreign key : -

Fungsi : Sebagai penyimpanan *data Server*.



Tabel 4.1 Samsat

No	Field	Data Type	Length	Description
1	id	Integer	-	Id <i>Server</i>
2	ip	Varchar	15	Ip <i>Server</i>
3	nama	Varchar	50	Nama <i>Server</i>
4	status	Varchar	10	Status <i>Server</i>
5	kapasitas	varchar	10	<i>Bandwidth Point Line</i>
6	lineon	Varchar	25	Waktu <i>Server Online</i>
7	Lineoff	Varchar	25	Waktu <i>Server Offline</i>
8	dex	Varchar	10	<i>Index Server</i>

### 4.3 Mengimplementasikan Sistem

Mengimplementasikan sistem merupakan tahap pengujian dimana desain sistem dapat berjalan dengan baik. Implementasi dilakukan oleh satu orang. Desain halaman yang telah dibuat cukup sesuai untuk mengimplementasikan sistem, sehingga tidak banyak membutuhkan perubahan.

#### 4.3.1 Spesifikasi Perangkat

Sistem yang digunakan untuk menjalankan aplikasi *Monitoring Network* terdiri dari *hardware* dan *software* pendukung. Adapun *hardware* dan *software* pendukung yang digunakan adalah sebagai berikut:

Spesifikasi *hardware* pendukung terdiri dari:

1. *Microprocessor Pentium III* atau yang lebih tinggi.
2. *Memory 1 GB RAM* atau yang lebih tinggi.
3. *10/100 Mbps Ethernet Network Interface Card*.
4. *Hardisk* minimal 20 GB.

Spesifikasi *Software* pendukung terdiri dari:

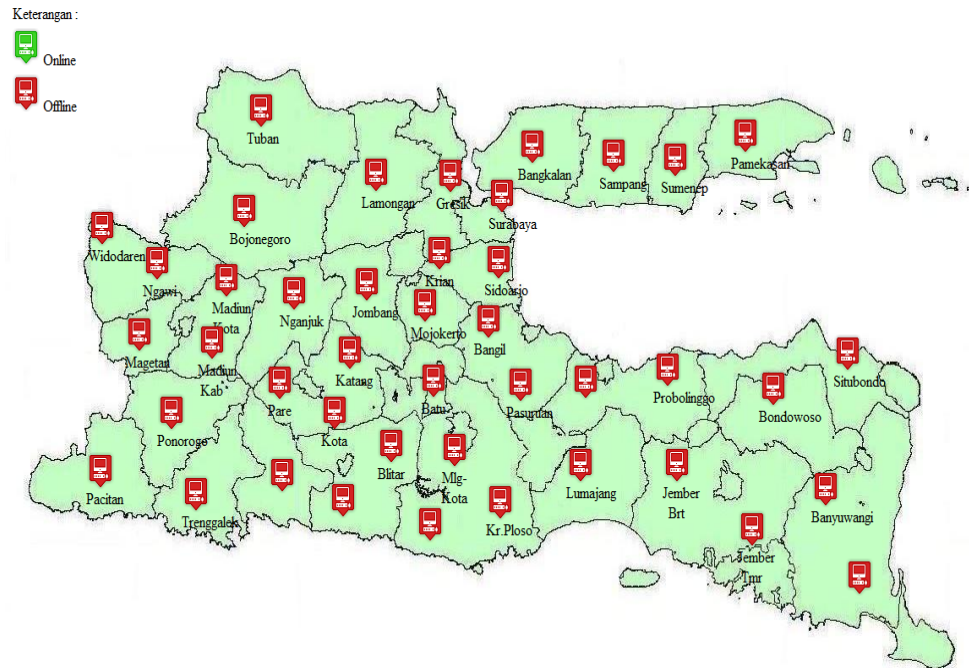
1. Sistem Operasi *Microsoft Windows 2000 Server/Pro, XP Profesional/Home Edition, Microsoft Windows Vista Basic, Premium, Bussines, Ultimate*, atau juga *Windows 7*.
2. *Web Browser* disarankan memakai *Mozilla Firefox*.
3. *XAMPP (PHP, Apache)*
4. *Adobe Dreamweaver CS 5*

#### 4.3.2 Penjelasan Penggunaan Program

Berikut adalah penjelasan penggunaan masing-masing halaman yang ada pada aplikasi *Monitoring Network* berbasis *web* pada Dinas Pendapatan Provinsi Jawa Timur.

##### 1. Halaman Utama

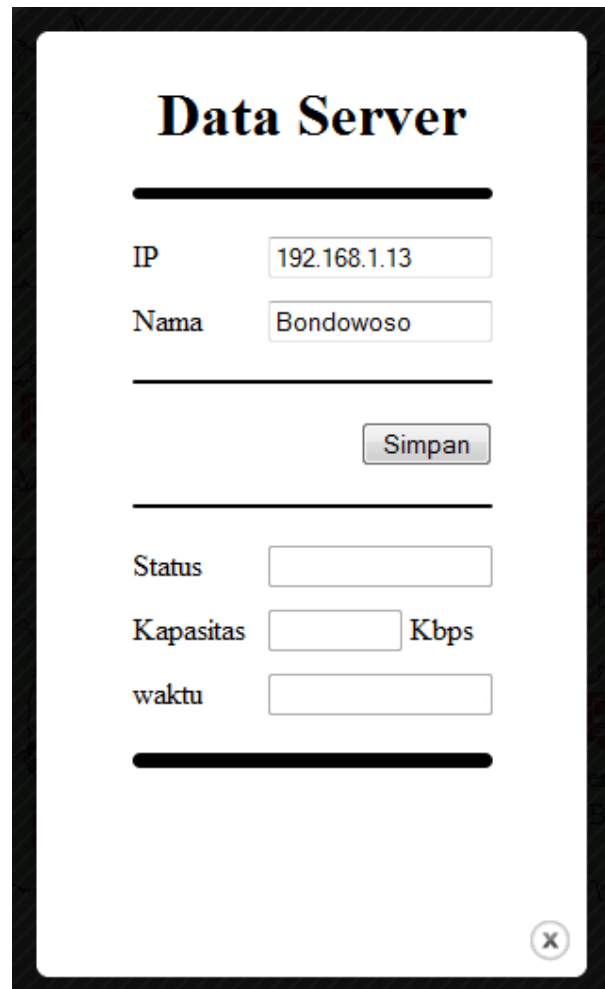
Halaman Utama ini digunakan untuk melihat status *Server* yang ada di seluruh Jawa Timur dengan melihat gambar *Server* yang berubah-ubah warna, seperti hijau dan merah. Gambar *Server* hijau mencerminkan bahwa *Server* dalam status *connect*, sedangkan gambar *Server* merah mencerminkan bahwa *Server* dalam status *disconnect*. Pada halaman utama ini gambar *Server* dapat di-klik sehingga masuk kedalam halaman penginputan *data Server*. Halaman utama dapat dilihat pada Gambar 4.7 di bawah ini.



Gambar 4.7 Halaman Utama Aplikasi *Monitoring Network*.

## 2. Halaman Penginputan *Data Server*

Halaman penginputan *data Server* ini digunakan untuk mengisi *data* setiap *Server*, *data* yang diisikan didalam *Form Data Server* hanya kolom IP dan Nama saja lalu tekan tombol simpan sehingga kita bisa mengetahui apakah *Server* tersebut dalam kondisi *connect* atau *disconnect*, dalam halaman ini juga kita dapat mengetahui kapasitas *bandwidth* setiap jaringan *Server* dan waktu *connect* atau *disconnect Server*. Halaman penginputan dapat dilihat pada Gambar 4.8 di bawah ini.



**Data Server**

---

IP

Nama

---

---

Status

Kapasitas  Kbps

waktu


---

Gambar 4.8 Halaman Penginputan *Data Server*.

### 3. Proses Halaman Penginputan *Data Server*

Pada halaman penginputan *data Server*, dilakukan pengecekan IP *Server* yang dimasukan apakah dalam status *connect* atau *disconnect*. Jika status *Server connect* akan menampilkan kapasitas *bandwidth* jaringan dalam ukuran Kbps dan jika status *disconnect* akan menampilkan kapasitas *bandwidth* jaringan 0 (nol) Kbps serta menampilkan waktu saat status *Server* dalam keadaan *connect* atau *disconnect*. Tidak hanya mengecek IP *Server* saja dalam proses ini juga menyimpan *data* yang ada di *form* penginputan *data Server* kedalam *database*

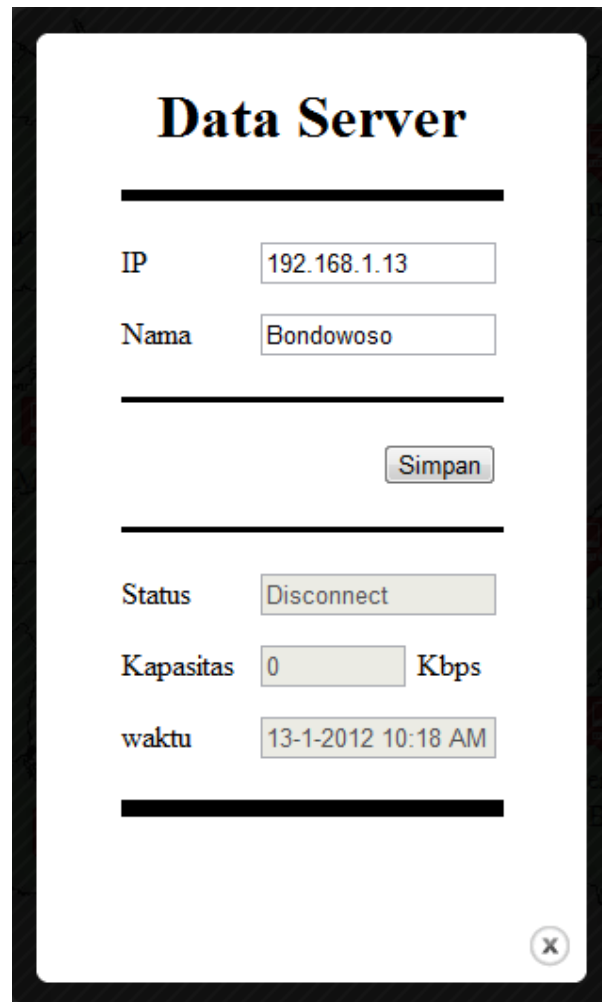
yang bernama Samsat. Dapat dilihat pada Gambar 4.9 dan Gambar 4.10 yang terdapat di bawah ini.



The image shows a web form titled "Data Server" with a black border. The form contains several input fields and a button. The fields are: "IP" with the value "192.168.1.13", "Nama" with the value "Bondowoso", "Status" with the value "Connect", "Kapasitas" with the value "5" and the unit "Kbps", and "waktu" with the value "13-1-2012 15:25 PM". There is a "Simpan" button below the "Nama" field. The form is divided into sections by horizontal lines. A close button (X) is located in the bottom right corner of the form area.

Data Server	
IP	192.168.1.13
Nama	Bondowoso
<input type="button" value="Simpan"/>	
Status	Connect
Kapasitas	5 Kbps
waktu	13-1-2012 15:25 PM

Gambar 4.9 Proses Penginputan *Data Server* Dalam Status *Connect*.



**Data Server**

---

IP

Nama

---

---

Status

Kapasitas  Kbps

waktu

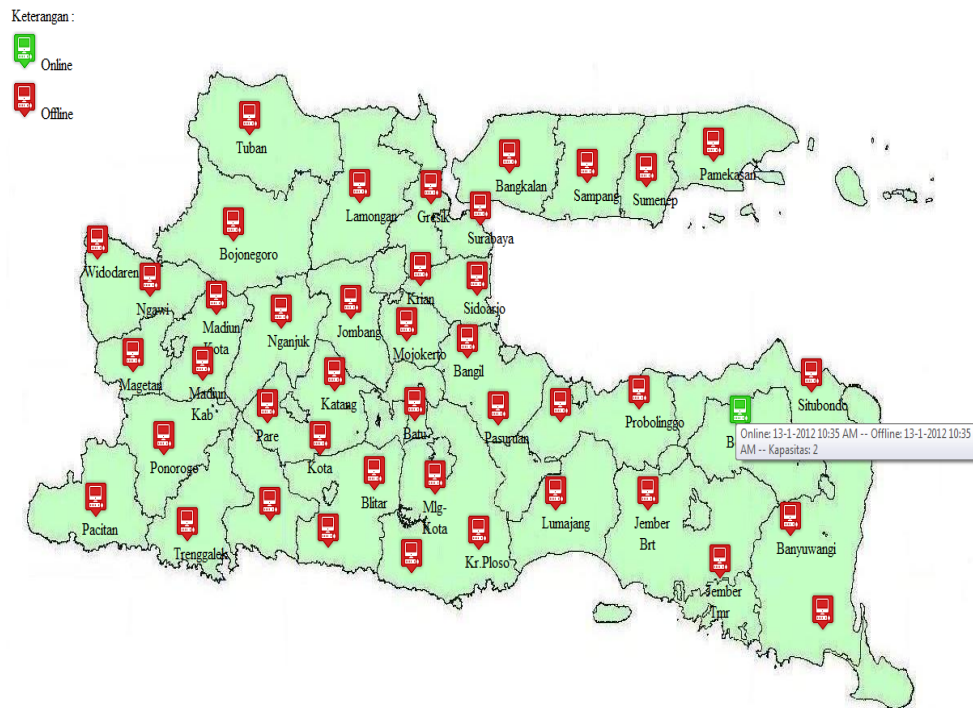
---

Gambar 4.10 Proses Penginputan *Data Server* Dalam Status *Disconnect*.

#### 4. *Output* Penginputan *Data Server*

*Output* dari penginputan *data Server* ini adalah akan merubah gambar yang terdapat di halaman utama yang awalnya gambar berwarna merah akan diganti berdasarkan status yang terdapat di *form* penginputan *data Server*. jika status *connect* maka gambar di halaman utama akan diganti menjadi hijau sedangkan jika status *disconnect* maka gambar di halaman utama akan diganti menjadi merah atau tetap seperti gambar awal di halaman utama tidak hanya

memberikan informasi status *server connect* atau *disconnect* tetapi juga memberikan informasi kapan waktu terakhir *online* dan *offline* serta memberikan informasi kapasitas *bandwidth* disetiap *server*. Dapat dilihat di Gambar 4.11 di bawah ini.



Gambar 4.11 *Output Penginputan Data Server.*